Product Structuring:
how could it help us in the future
and why isn't it helping us now?

Wilfred van der Vegte

The Problem To Overcome: Properties You've Never Asked For

During the development of a product, designers unconsciously make decisions which they do not know the consequences of. The following simplified example is an observation from everyday design practice.

A PCB is to be secured in an enclosure.
Solution: connect with four screws into hollow bosses.
→ problem solved

but ... other solutions have not been considered:
can it be done with just one screw?
can it be done with snap fits?

Eventually the product, containing lots of PCB's, turns out not to satisfy the cost target, so that costly design changes have to be made.

In this case, apparently, the designer did not realize that he made two *decisions* by securing the PCB with threaded tightening, and by using *four screws* to do so.

How To Fight Properties You've Never Asked For?

In the worst case, the negative consequences of some of the features not considered arise in production or even during product use.
Having learned from past experiences, more and more companies are trying to fight unwanted properties by *intermittently* subjecting the design to so called design review sessions. Such sessions are often linked to development phases or milestones. In this way, the amount of work to be undone in case of a design change is kept within certain limits.

It would be even better, however, if all relevant features can be considered continuously. In this way, we could make all design iterations incremental, undetectable and thus of negligible influence on product development costs. This implies, that we will not only have a continuously changing drawing but also some file, or call it *additional product structure,* that changes with the drawing and which is probably linked to it, carrying all other relevant information which the designer could possibly enter, such as:

- intended process steps in the manufacturing of each part;
- intended process steps in the assembly of the product (including intended hierarchic division)
- assumed service parts and their service characteristics (service frequency, to be replaced/repaired/cleaned/...)
- inter-part relationships (A connected to B with snap fit; A covers C, etc.)
- part (dis)assembly features (handling characteristics, presence of chamfers etc.)
- failure modes of parts and part groups, causes and effects of failures; occurrence, severity and detection levels
- material types and colours
- ...

Product structure language: requirements

From the above, and from practical design experience, the following requirements for a universal product structure language can be gathered:

- easy to use and understand for a design team;
- linkable to (CAD) drawing information and part lists;
- extendable with design parameters not yet considered important;
- capable of handling incomplete sets of data, thus enabling application at early design stages;

(this list is not exhaustive)

Product structure processing: output requirements

Processing of the product structure should provide:

- real time confrontation with the consequences of each design decision (even if this decision is not recognized as a decision);
- an overview of data yet to be entered
- an overview of bottle necks in manufacturing, assembly, disassembly, service, safety, reliability, ...;
- priorities for improvement.

(this list is not exhaustive)